Jurnal Informatika dan Rekayasa Perangkat Lunak ISSN 2656-2855
Vol. 6, No. 1, Oktober 2024, Hal. 281-288 e-ISSN 2685-5518

Rancang Bangun Aplikasi Blog Liora Berbasis Next.js dengan Desain Responsif dan Implementasi PWA pada Platform Android

Annas Aulia Rahman [1]

Program Studi Teknik Informatika, Universitas Wahid Hasyim
annasauliarahman04@gmail.com

Abstrak

Penelitian ini membahas tentang perancangan dan pembangunan aplikasi blog bernama Liora menggunakan framework Next.js dengan desain responsif berbasis Tailwind CSS, serta penerapan teknologi Progressive Web App (PWA) agar dapat dijalankan di perangkat Android melalui Android Studio. Latar belakang penelitian ini adalah kebutuhan akan platform blog yang ringan, cepat, dan mudah diakses di berbagai perangkat tanpa bergantung pada toko aplikasi. Pengembangan dilakukan melalui beberapa tahap, yaitu analisis kebutuhan, perancangan antarmuka, implementasi sistem menggunakan Next.js dan Tailwind CSS, serta pengujian performa dan kompatibilitas aplikasi. Hasil penelitian menunjukkan bahwa penggunaan Next.js dan Tailwind CSS mampu meningkatkan efisiensi dalam proses pengembangan antarmuka dan menghasilkan performa yang optimal, sedangkan penerapan PWA memberikan pengalaman pengguna yang menyerupai aplikasi native dan memperluas jangkauan akses karena dapat dijalankan langsung melalui browser maupun diinstal sebagai aplikasi Android. Dengan demikian, aplikasi Liora dapat menjadi solusi efektif dalam pengembangan blog modern yang responsif, ringan, dan mudah diakses di berbagai platform.

Kata kunci: *nextjs, PWA, Android, Android Studio, Blog*

Abstract

This research discusses the design and development of a blog application called Liora using the Next.js framework with a responsive design based on Tailwind CSS, and the implementation of Progressive Web App (PWA) technology to make it run on Android devices through Android Studio. The background of this research lies in the need for a lightweight, fast, and easily accessible blog platform that can run on various devices without relying on traditional app stores. The development process includes several stages, such as requirements analysis, interface design, system implementation using Next.js and Tailwind CSS, and performance and compatibility testing. The results show that the use of Next.js and Tailwind CSS improves efficiency in interface development and produces optimal performance, while the implementation of PWA provides a user experience similar to native applications and expands accessibility since it can be accessed directly through a browser or installed as an Android application. Therefore, the Liora application can serve as an effective solution for developing a modern, responsive, and easily accessible blog platform across multiple platforms.

Kata kunci: *Nextjs, PWA, Android, Android Studio, Blog*

PENDAHULUAN

Keberadaan berbagai alat modern untuk teknologi web mempercepat pembangunan aplikasi web karena menjadi lebih interaktif hingga responsif. Next.js termasuk framework web modern pembangun aplikasi yang dibangun menggunakan React dengan kelebihan rendering sisi server dan generasi situs statis. Next.js memudahkan para pembangun aplikasi web dengan performa tinggi dan sistem pengodean yang baik. Merender banyak konten blog dapat membantu pembaca

Sekarang ini, banyak pembaca yang menggunakan smartphone untuk mengakses konten dan mengoptimalkan pengalaman pengguna sangatlah penting. Maka dari itu, pengembangan aplikasi blog yang responsif mobile menjadi salah satu tantangan pengembangan web.

Aplikasi Liora adalah proyek blog yang dibangun dengan fokus pada desain responsif untuk ukuran layar mobile, di Next.js. Proyek ini dirancang untuk dijalankan sebagai aplikasi web, dan dirancang untuk dijalankan sebagai Aplikasi Web Progresif (PWA). Liora memungkinkan pengguna untuk menginstalnya sebagai aplikasi native di perangkat Android mereka. Dengan mengadopsi PWA, Liora dapat dibangun sebagai file APK melalui Android Studio, meningkatkan pengalaman pengguna dalam hal kecepatan akses, kemudahan penggunaan, dan ketersediaan offline.

Tujuan dari penelitian ini adalah untuk merancang dan membangun aplikasi blog Liora berbasis Next.js yang responsif, serta mengimplementasikannya dalam bentuk PWA agar dapat dijalankan pada platform Android. Hasil dari penelitian ini diharapkan dapat memberikan kontribusi dalam pengembangan aplikasi web modern yang efisien, responsif, dan memiliki kemampuan lintas platform antara web dan mobile.

TINJAUAN PUSTAKA

2.1. Blog

Blog adalah sius web yang berisi tulisan atau artikel yang diperbarui secara berkala oleh penulis atau pengguna. Umumnya blog digunakan untuk membagikan informasi, pengalaman, atau opini dalam bentuk teks, gambar, maupun video. Blog juga menjadi media digital yang mudah digunakan karena pengguna dapat mengelola dan mempublikasikan konten tanpa memerlukan pengetahuan teknis yang rumit.

Selain itu, blog juga berperan penting dalam dunia digital sebagai sarana personal branding, edukasi, serta penyebaran informasi secara cepat. Dengan perkembangan teknologi web modern, blog kini dapat dibuat lebih interaktif dan responsif menggunakan berbagai framework serta dapat diakses dengan mudah melalui perangkat mobile.

2.2. Nextjs

Next.js adalah salah satu framework JavaScript berbasis React yang digunakan untuk membangun aplikasi web modern. Framework ini mendukung fitur server-side rendering (SSR) dan static site generation (SSG) yang membuat performa website menjadi lebih cepat dan efisien. Dengan kemampuan tersebut, Next.js memungkinkan halaman web dimuat lebih ringan dan mudah diindeks oleh mesin pencari.

Selain itu, Next.js juga memiliki sistem pengaturan rute (routing) yang sederhana dan terstruktur, sehingga pengembang dapat membuat halaman baru hanya dengan menambahkan file di dalam folder tertentu. Fitur ini membuat proses pengembangan menjadi lebih cepat dan mudah dikelola. Next.js juga terintegrasi dengan berbagai layanan API dan mendukung pengambilan data secara dinamis, sehingga cocok digunakan untuk aplikasi web interaktif dan berskala besar.

Dalam penelitian ini, Next.js digunakan sebagai kerangka utama dalam pembangunan aplikasi blog Liora. Framework ini dipilih karena stabil, memiliki komunitas besar, serta mendukung penerapan desain responsif melalui integrasi dengan Tailwind CSS. Dengan menggunakan Next.js, aplikasi blog Liora dapat berjalan dengan baik di berbagai perangkat dan tetap memiliki performa tinggi ketika diakses melalui browser maupun platform Android berbasis PWA.

2.3. Tailwindcss

Tailwind CSS adalah framework CSS yang berfokus pada penggunaan kelas-kelas utilitas untuk mempercepat proses pembuatan tampilan antarmuka. Dengan Tailwind, pengembang dapat langsung menerapkan gaya seperti warna, ukuran, margin, padding, dan tata letak hanya dengan menambahkan kelas bawaan pada elemen HTML tanpa perlu menulis kode CSS secara manual. Hal ini membuat proses pengembangan menjadi lebih cepat, fleksibel, dan mudah disesuaikan sesuai kebutuhan desain.

Selain itu, Tailwind CSS mendukung fitur responsive design yang memungkinkan tampilan aplikasi menyesuaikan dengan berbagai ukuran layar perangkat seperti smartphone, tablet, dan desktop. Dengan pendekatan berbasis utilitas ini, desain antarmuka menjadi lebih konsisten dan efisien, sehingga sangat membantu dalam pembuatan aplikasi blog Liora yang menekankan tampilan modern dan responsif.

2.4. PWA

Progressive Web App atau PWA adalah teknologi yang membuat website bisa berfungsi seperti aplikasi di handphone. Dengan PWA, pengguna bisa membuka aplikasi langsung dari layar utama tanpa harus mengunduhnya dari Play Store. Selain itu, PWA juga bisa digunakan walau sedang tidak ada koneksi internet dan memiliki tampilan yang cepat serta responsif.

Penerapan PWA memberikan banyak keuntungan bagi pengembang maupun pengguna, karena teknologi ini menggabungkan kelebihan dari website dan aplikasi native. Dari sisi pengembang, PWA lebih mudah di-deploy dan di-update tanpa perlu melalui proses distribusi aplikasi di toko resmi. Sementara dari sisi pengguna, PWA menawarkan pengalaman penggunaan yang konsisten, cepat, dan ringan, bahkan di perangkat dengan spesifikasi rendah. Hal ini menjadikan PWA sebagai solusi yang efisien untuk meningkatkan aksesibilitas dan performa aplikasi berbasis web di berbagai platform.

2.5. Android

Android adalah sistem operasi berbasis Linux yang digunakan pada perangkat mobile seperti smartphone dan tablet. Android dikembangkan oleh Google dan menjadi sistem operasi paling populer di dunia karena bersifat *open source* dan mendukung banyak perangkat dari berbagai merek.

Android memudahkan pengembang untuk membuat dan mengembangkan aplikasi melalui berbagai bahasa pemrograman seperti Java, Kotlin, maupun framework berbasis web yang dikonversi menjadi aplikasi Android. Dalam penelitian ini, Android digunakan sebagai platform untuk menjalankan aplikasi blog Liora yang dibangun dengan teknologi PWA, sehingga aplikasi dapat diinstal dan digunakan layaknya aplikasi native di Android.

2.6. Android Studio

Android Studio adalah aplikasi yang digunakan untuk membuat dan membangun aplikasi Android. Platform ini menyediakan berbagai alat untuk menulis kode, mendesain tampilan, dan menguji aplikasi langsung di emulator atau perangkat Android. Dengan Android Studio, pengembang dapat mengubah website seperti PWA menjadi aplikasi yang bisa diinstal di handphone dengan mudah.

Selain itu, Android Studio juga memiliki fitur-fitur yang membantu proses pembuatan aplikasi, seperti alat untuk melihat dan memperbaiki error, menguji aplikasi di berbagai ukuran layar, serta memantau performa aplikasi agar tetap ringan dan cepat. Dengan adanya fitur-fitur tersebut, pengembang bisa lebih mudah memastikan bahwa aplikasi yang dibuat berjalan dengan baik di berbagai jenis perangkat Android.

METODE PENELITIAN

3.1. Jenis Penelitian

Penelitian ini menggunakan metode rekayasa perangkat lunak dengan pendekatan pengembangan langsung berdasarkan kebutuhan pengguna. Tujuan penelitian ini adalah untuk merancang dan membangun aplikasi blog *Liora* yang responsif dan dapat dijalankan sebagai *Progressive Web App (PWA)* pada perangkat Android.

Metode penelitian ini mencakup beberapa tahapan utama, yaitu perencanaan, perancangan antarmuka (UI/UX), implementasi kode program, pengujian, serta proses build aplikasi menjadi format yang dapat dijalankan di perangkat Android melalui Android Studio. Setiap tahapan dilakukan secara bertahap untuk memastikan aplikasi berjalan dengan baik, tampilan antarmuka sesuai kebutuhan pengguna, dan fungsi-fungsi utama seperti pembuatan serta penayangan artikel dapat digunakan dengan optimal. Pendekatan ini memungkinkan proses pengembangan yang fleksibel terhadap perubahan kebutuhan selama proses perancangan berlangsung.

3.2. Tahapan Penelitian

Tahapan dalam penelitian ini dilakukan melalui empat langkah utama sebagai berikut:

1. Perencanaan, Tahap awal dilakukan penentuan ide dan konsep aplikasi yang akan dibuat, yaitu blog *Liora*. Dalam tahap ini ditentukan tujuan pembuatan, target pengguna, serta fitur-fitur utama yang akan diterapkan seperti sistem posting, tampilan responsif, dan dukungan PWA.
2. Perancangan design, perancangan antarmuka pengguna (*User Interface*) dan pengalaman pengguna (*User Experience*). Desain dibuat agar tampilan blog terlihat menarik, mudah digunakan, dan tetap nyaman diakses melalui berbagai ukuran layar, terutama perangkat mobile.
3. Implementasi, Tahap ini merupakan proses pembuatan aplikasi menggunakan *framework* Next.js untuk logika dan struktur halaman, serta Tailwind CSS untuk desain tampilan agar lebih cepat dan konsisten. PWA diterapkan agar aplikasi dapat diakses seperti aplikasi Android tanpa harus diunduh dari Play Store.
4. Debugging, Proses debugging dilakukan untuk menemukan dan memperbaiki kesalahan yang terjadi selama pengembangan aplikasi blog Liora. Beberapa error yang muncul antara lain pada tampilan responsif dan konfigurasi PWA. Proses perbaikan dilakukan dengan memanfaatkan fitur konsol di browser, log pada terminal Visual Studio Code, serta pengujian langsung di emulator Android Studio. Dengan debugging yang dilakukan secara berkala, aplikasi dapat berjalan dengan baik tanpa error dan sesuai dengan fungsi yang diharapkan.
5. Build Aplikasi, Setelah aplikasi selesai dan berfungsi dengan baik di browser, tahap berikutnya adalah melakukan *build* menggunakan Android Studio. Proses ini mengubah hasil proyek Next.js yang telah diimplementasikan PWA menjadi file aplikasi (APK) sehingga dapat diinstal dan dijalankan langsung di perangkat Android.

3.3. Kebutuhan Perangkat lunak

Dalam proses pengembangan aplikasi blog *Liora*, diperlukan beberapa perangkat lunak pendukung agar proses perancangan, pembuatan, dan pengujian dapat berjalan dengan baik. Adapun perangkat lunak yang digunakan adalah Visual Studio Code, Next.js , Tailwind CSS, Google Chrome,Git, Terminal, NPM, Android Studio.

3.4. Kebutuhan Perangkat Keras

Kebutuhan perangkat keras yang dibutuhkan unutk mengimplementasikan *code* program menjadi aplikasi android (.apk) adalah laptop penulis yaitu LENOVO THINKPAD L420 dengan RAM 8GB, I5 2520, SSD 120GB.

HASIL DAN PEMBAHASAN

4.1. Hasil dan Analisa

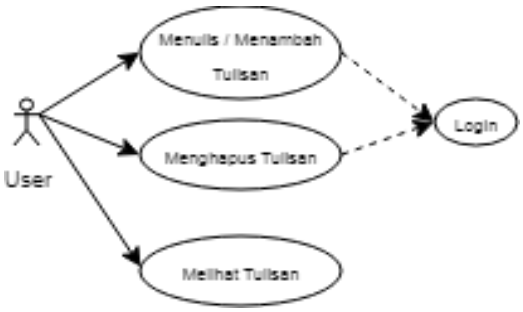
Hasil analisa dari aplikasi ini adalah sebuah aplikasi blog bernama *Liora* yang dibangun menggunakan *framework* Next.js dengan tampilan responsif melalui Tailwind CSS, serta dilengkapi fitur *Progressive Web App (PWA)* agar dapat dijalankan di perangkat Android. Aplikasi ini memungkinkan pengguna untuk membaca dan menulis artikel dengan tampilan yang sederhana dan nyaman di device mobile.

Yang memiliki halaman utama, yaitu :

a. Halaman Beranda, yang menampilkan daftar artikel
b. Halaman Detail Artikel, yang menampilkan isi dari artikel
c. Halaman *Publish* artikel, yang memungkinkan pengguna unutk menulis dan mempublikasikan tulisannya
d. Halaman Profile, menampilkan informasi pengguna
e. Halaman Authentikasi, untuk pengguna melakukan *Login* dalam aplikasi

4.2. *Hasil Flow Aplikasi*

Use case dalam aplikasi ini terdapat 1 aktor yaitu pengguna sebagai penulis dan juga sekaligus sebagai pembaca. *Use case* aplikasi sebagai berikut.



Gambar 1 Flow dan use case aplikasi

Pada diagram use case di atas, alur sistem menunjukkan interaksi antara pengguna) dengan aplikasi blog Liora. Pengguna dapat melakukan beberapa aktivitas utama, yaitu menulis atau menambah tulisan, menghapus tulisan, dan melihat tulisan. Sebelum dapat menulis atau menambah tulisan, pengguna harus melakukan proses login terlebih dahulu sebagai bentuk autentikasi agar sistem dapat mengenali identitas pengguna. Setelah berhasil masuk, pengguna dapat membuat tulisan baru atau menambah tulisan yang sudah ada. Selain itu, pengguna juga memiliki hak untuk menghapus tulisan yang telah dibuat, serta dapat

melihat seluruh tulisan yang tersedia di dalam aplikasi. Dengan demikian, diagram tersebut menggambarkan alur utama yang menjelaskan bagaimana pengguna berinteraksi dengan sistem blog secara umum.

4.3. Hasil Implementasi

Hasil implementasi dari sistem blog Liora menunjukkan bahwa seluruh fitur utama telah berjalan dengan baik sesuai rancangan. Pengguna dapat melakukan proses login menggunakan akun Google, kemudian menulis, menghapus, dan melihat tulisan secara responsif melalui tampilan antarmuka yang sederhana dan mudah digunakan. Sistem berhasil diintegrasikan dengan baik pada aplikasi, serta dapat di jalankan di perangkat android.

Selain itu, hasil pengujian menunjukkan bahwa performa aplikasi berjalan dengan stabil dan responsif pada berbagai ukuran layar, mulai dari smartphone hingga tablet. Fitur Progressive Web App (PWA) juga berfungsi dengan baik, memungkinkan pengguna untuk menambahkan aplikasi ke layar utama dan mengaksesnya tanpa koneksi internet setelah halaman dimuat sebelumnya. Hal ini membuktikan bahwa penerapan PWA pada aplikasi blog Liora berhasil meningkatkan pengalaman pengguna dengan memberikan kemudahan akses layaknya aplikasi native.

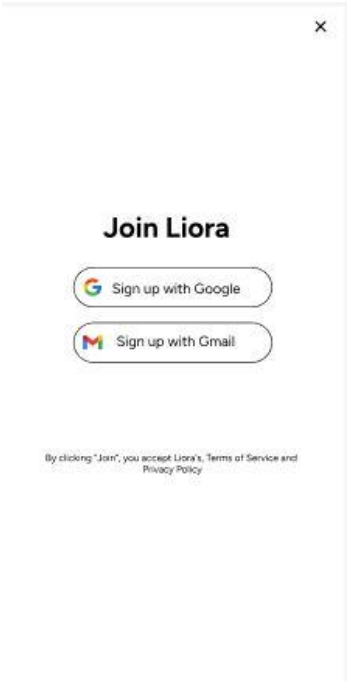
Dari sisi tampilan dan pengalaman pengguna (UI/UX), desain yang diterapkan menggunakan Tailwind CSS memberikan hasil yang konsisten dan menarik secara visual. Desain responsif memastikan setiap elemen tata letak menyesuaikan ukuran layar dengan baik, tanpa mengurangi kenyamanan membaca atau menulis artikel. Dengan demikian, aplikasi blog Liora tidak hanya berfungsi dengan optimal secara teknis, tetapi juga memberikan pengalaman pengguna yang baik dan sesuai dengan tujuan awal penelitian, yaitu menciptakan aplikasi blog modern yang efisien, ringan, dan mudah digunakan di berbagai perangkat.

Pada Gambar 2.1 Tampilan Homescreen, Mencakup heading pembuka aplikasi

Gambar 2.1 Halaman Homescreen

menampilkan tampilan halaman awal (landing page) aplikasi Liora. Pada bagian atas terdapat logo dan tombol Get started, sedangkan di tengah terdapat teks utama Words, Thoughts, and Stories. yang menggambarkan tema aplikasi sebagai media berbagi cerita dan pemikiran.

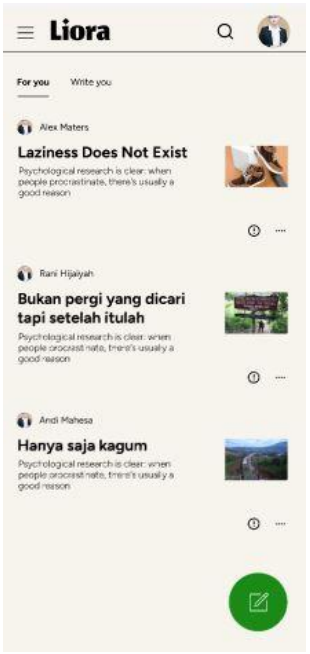
Pada Gambar 4.2 Tampilan Authentikasi yang hanya menggunakan Gmail



Gambar 2.2 Halaman Authentikasi

Gambar tersebut menampilkan halaman pendaftaran aplikasi Liora. Pada halaman ini, pengguna dapat mendaftar menggunakan akun Google atau Gmail melalui dua tombol yang tersedia. Desainnya sederhana dan minimalis agar pengguna dapat dengan mudah bergabung ke dalam aplikasi.

Pada Gambar 2.3 Tampilan Homepage, Halaman yang menampilkan semua data tulisan



Gambar 2.3. Halaman Homepage

Gambar tersebut menampilkan halaman utama aplikasi Liora setelah pengguna masuk. Pada halaman ini terdapat daftar postingan blog dari beberapa penulis lengkap dengan judul, nama penulis, dan gambar pendukung. Tersedia juga menu navigasi di bagian atas serta tombol berbentuk pena di kanan bawah untuk membuat tulisan baru. Tampilan ini dirancang sederhana dan mudah digunakan agar pengguna dapat membaca atau menulis artikel dengan nyaman.

Pada Gambar 2.4 Tampilan isi artikel, yang memuat keseluruhan dari tulisan

Gambar 2.4 Halaman isi Artikel

Gambar tersebut menampilkan halaman pembacaan artikel di aplikasi Liora. Halaman ini berisi judul, nama penulis, tanggal publikasi, gambar utama, serta isi artikel yang dapat dibaca oleh pengguna. Tampilan dibuat sederhana agar fokus pada konten tulisan.

Pada Gambar 2.5 Tampilan Profile akun, yang mencakup informasi user

Gambar 2.5 Halaman Profile Akun

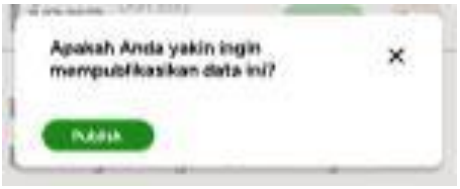
Gambar tersebut menunjukkan halaman profil pengguna pada aplikasi liora. Dan terdapat foto profil, nama pengguna, serta daftar tulisan yang telah dibuat. Halaman ini berfungsi untuk menampilkan hasil karya pengguna secara teratur dan mudah dibaca.

Pada Gambar 2.6 Tampilan Form Menulis, untuk penulis

Gambar 2.6 Halaman Form Tulis Artikel

halaman itu untuk membuat cerita baru pada aplikasi liora. Di halaman ini pengguna dapat menulis judul, deskripsi singkat tentang cerita, menambahkan gambar dengan menekan tanda plus, serta menulis isi cerita yang ingin dibagikan.

Pada Gambar 2.7 Tampilan untuk mempublish tulisan



Gambar 2.7 Notifikasi publish Tulisan

5. Kesimpulan

Berdasarkan hasil penelitian dan pembuatan aplikasi blog Liora, dapat disimpulkan bahwa proses pengembangan menggunakan framework Next.js dan Tailwind CSS mampu menghasilkan aplikasi blog yang responsif, cepat, dan mudah diakses di berbagai perangkat.

Penerapan Progressive Web App (PWA pada aplikasi ini juga berhasil membuat Liora dapat diinstal dan dijalankan langsung di perangkat Android tanpa melalui Play Store, sehingga meningkatkan kemudahan dan pengalaman pengguna.

Secara keseluruhan, aplikasi blog Liora berhasil memenuhi tujuan penelitian, yaitu merancang dan membangun aplikasi blog modern yang responsif, ringan, dan dapat diakses baik melalui web maupun sebagai aplikasi Android.